Data-Driven Machine Learning Approach for Human Action Recognition Using Skeleton and Optical Flow

Yen-Ting Lee[1], Thitinun Pengying[2], Sule Yildirim Yayilgan[3] and Ogerta Elezaj[4]

[1] Norwegian University of Science and Technology, Teknologivegen, 22, 2815 Gjøvik, Norway
leeyt1377@gmail.com
[2] Norwegian University of Science and Technology, Teknologivegen, 22, 2815 Gjøvik, Norway
thitinup4@gmail.com
[3] Norwegian University of Science and Technology, Teknologivegen, 22, 2815 Gjøvik, Norway
sule.yildirim@ntnu.no
[4] Norwegian University of Science and Technology, Teknologivegen, 22, 2815 Gjøvik, Norway
ogerta.elezaj@ntnu.no

Abstract. Human action recognition is a very challenging problem due to numerous variations in each body part. In this paper, we propose a method for extracting optical flow information from skeleton data to address the problem of body part movement variation in human action recognition. The additional arm part information was also analyzed how valuable it was. Then, different machine learning methods are applied such as k-Nearest Neighbors (KNNs) and deep learning to recognize human actions on the UTKinect-Action 3D dataset. We then design and train different KNNs models and Deep Convolutional Neural Networks (D-CNNs) on the obtained image and classify them into classes. Different numbers of features from histogram data collection are used to recognize 10 categories of human actions. The best accuracy we obtained is about 88%. The proposed method had improved accuracy to almost 97% with only 5 classes. These features are representative to describe the human action and recognition which does not rely on plenty of training data. Results of experiments show that using deep learning can lead to better classification accuracy.

Keywords: Human action recognition, Skeleton, Optical flow, KNNs, Deep learning.

1 Introduction

Action classification is one of the challenging problems because different persons perform the same action differently which leads to variation in body part movement and variation from frame to frame in videos. However, this problem caught attention from various application areas e.g. video surveillance, healthcare system and robotics.

With the development of depth map, researchers began to apply it to human action recognition since it provides 3D data instead of 2D. In particular, depth data based human skeleton tracking technology achieves outstanding precision and stimulates the research on human action recognition to use skeleton data [1]. So far, skeleton-based action recognition has been widely studied where input data is 3D skeleton data (RGB images and depth map) in general.

Moreover, different approaches have been utilized for higher efficiency. Histograms of 3D joint locations (HOJ3D) extracted from Kinect 3D skeletal joint locations by using Linear Discriminant Analysis (LDA) projection to represent poses and then modelling by discrete hidden Markov models (HMMs) gives more than 90% accuracy [2]. Using skeleton data extracted via RGBD sensors has shown promising classification results on five different datasets where clustering and Support Vector Machine (SVM) classification has been used [3]. Converting dissimilarity space with 3D skeleton joints to torso-PCA- frame (TPCAF) coordinate system have shown equivalent performance with other methods on available datasets [4]. However, skeleton-based methods have two major obstacles: inaccurate skeleton estimation and intra-class variation in human actions. Nowadays, there are many open source skeleton extraction algorithms available. OpenPose is one of the well-known algorithms that has been proposed to detect 2D pose of multiple people in images or videos with great performance [5]. DRPose3D is another algorithm that used depth information together with 2D human joint locations and Convolutional Neural Networks (CNNs). It exceeded state-of-the-art method (KNNs) on Human3.6M dataset [6].

As deep learning gains more attention recently, there are many poses estimation-based research on it. Some solved the action classification problem by using CNNs on the MPII Human Pose dataset. Both [7] and [8] achieve state-of-the-art accuracy by using multiple stacked hourglass modules and combining heatmaps and regression respectively. Even some researchers have introduced unsupervised learning for human action recognition and use generative adversarial networks which has discriminator for this purpose to improve accuracy and let the generator learn more conceivable features for pose estimation. The first network improvised two discriminators and a multi-task pose generator which is robust to occlusions and human body distortion while the second network possesses identical discriminator and generator with the aid of heatmaps that help improve the prediction accuracy [9-10].

Skeleton-based action recognition can also be solved by using the deep learning approach. Inputting 3D skeleton data to CNNs with Long Short Term Memory (LSTM) regularization improves the accuracy from the-state-of-the-art method [11]. Separating the skeleton data into five parts before entering it into each subnet and hierarchically merging body parts back to the output layer of the recurrent neural network (RNN) accomplished both good performance and low computation time [12]. Besides the skeleton-based approach, videos can also be used as input data for training. Video representations learned by CNNs with long-term temporal convolutions (LTC) with flow help improve the performance over RGB and results in accurate optical flow estimation with a superior performance than state-of-the-art [13].

Optical flow is a major feature used in motion estimation alone or together with other aids. Spatial and temporal CNNs architecture trained on optical flow achieved good performance on the UCF-101 dataset [14]. Training only Farneback optical flow from RGB visualizations with CNNs on the KTH dataset showed assuring results. That is, the accuracy is good but still does not reach state of the art from using only a single feature [15]. In addition, applying neural networks enhance the performance of the flow. Fine tuning the different flow algorithms showed that action recognition accuracy correlates with flow near the boundary and small movements [16]. Using spatial gradients of feature maps and temporal gradients obtained from different frames as a feature to train the network have achieved very high accuracy on UCF-101 dataset [17].

Using optical flow as a feature with KNNs and Principal Component Analysis (PCA) achieved 100% accuracy on the KTH dataset [18]. On the other hand, KNNs have shown slight improvement of the accuracy with SVM than SVM alone on Wizeman dataset [19]. Based on the review provided so far, it is clear that optical flow helps obtain better results than state of the art and hence in this paper, we propose to train KNNs and deep learning with optical flow data extracted from skeleton data to classify human actions in videos with human activities.

This paper is organized as follows. Section 2 describes our methodology. Section 3 introduces the dataset and discusses the experimental results. Section 4 concludes the paper and also presents the future works.

2 Methodology

Fig. 1 illustrates the overall process of our method. In this section, we will provide an explanation of the processes in our method.

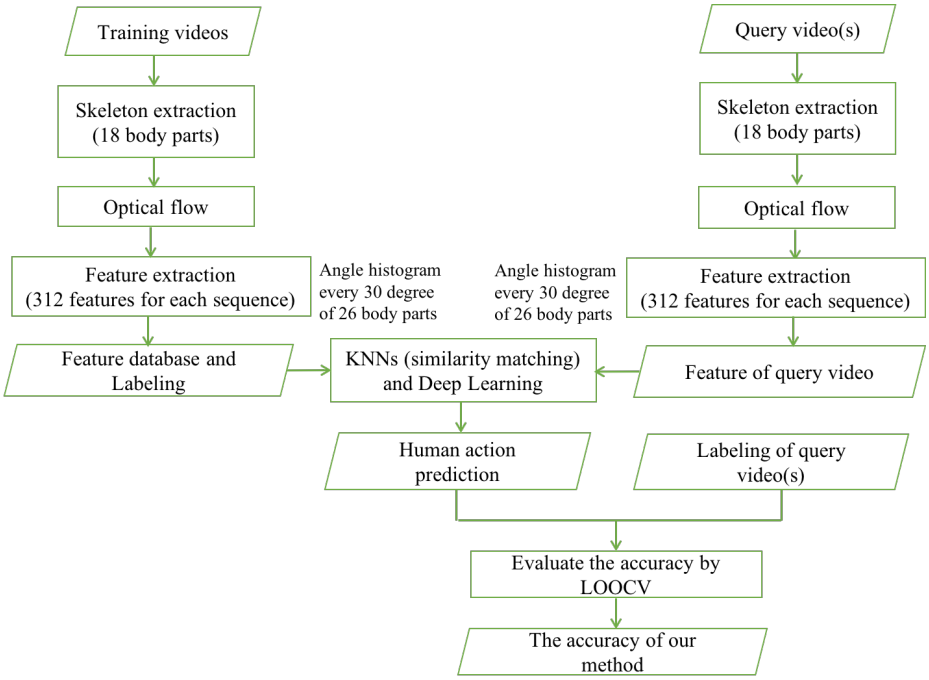

Fig. 1. Flowchart of our method.

2.1 Skeleton

In this project, OpenPose [5, 20, 21] is used to extract skeleton data from the frames. OpenPose is one of the open sources available (https://github.com/CMU-Perceptual-Computing-Lab/openpose) and it is the first real-time multi-person system to jointly

detect human body, hand, facial, and foot keypoints (in total 135 keypoints) on single image. It efficiently detects the 2D pose of multiple people in an image by using a non-parametric representation, referred to as Part Affinity Fields (PAFs) to learn to associate body parts with individuals in the image [5].

In our experiments, OpenPose represents a person by 25 body parts with 18 different colors. These 18 colors are regarded as the key body parts for human action recognition. Fig. 2 shows the *sit down* example and its skeleton result, the key body parts contain 18 different colors where the lower leg and foot are illustrated by the same color. In this paper, the input data of OpenPose is RGB images and it renders the skeletons on a black background as the output data to eliminate the undesired influence from the background. The user can modify the setting of OpenPose to have a different combination of key points in the output results. For example, different output format and keypoint ordering, body skeleton output, face keypoint output, hand keypoint output, and so on.

(A) Original RGB image (B) Skeleton extraction by OpenPose

Fig. 2. The result of skeleton by OpenPose.

2.2 Optical Flow

The extracted skeleton images are combined back to the .avi video by ImageJ [22] with 7 frames/second and converted to a .mov file which is compatible with Matlab in the Macintosh operating system.

Then, each frame is converted to grayscale before estimating the optical flow from the consecutive frames using the Farneback method. The result of the optical flow extraction process is shown in Fig. 3 where the left one displays the skeleton data obtained from OpenPose in the video of *carry* category and the right represents the optical flow of the video by the blue arrows. The larger the arrow is, the greater the movement of that part.

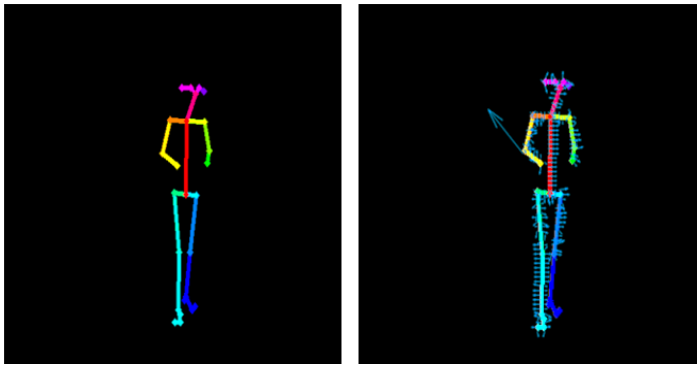

Fig. 3. Optical flow procedure: Left - original skeleton data and Right - optical flow.

After that all flows are collected from all videos, but this optical flow combines four properties: Vx, Vy, Orientation and Magnitude where the first two are the X and Y components of velocity respectively and the last two are the phase angles in radians and the magnitude of optical flow in 480x640 pixels. If all the flow information is exploited, the features' size will be larger and further extend the complexity of training. Also Orientation and Magnitude are more meaningful so they are chosen to use as the features.

Every frame of the video is extracted into different parts by using 18 color pixels but the size of the body in the image varied, therefore, the centroid of each part is found to represent the overall. The centroids of all 18 different parts can be seen in green compared to the whole skeleton in purple in Fig. 4. Then, the mean values of orientation and magnitude of each part of the continuous frames in each video are calculated as the features. Each video is composed of 18 orientation points and followed by another 18 magnitude points as features. As arm movements are very essential to distinguish between actions like *wave hands and clap hands*, two left and right points to the centroid of 4 arm parts are collected in order to get both orientation and magnitude for additional information. The comparison of 18 and 26 body parts will be demonstrated.

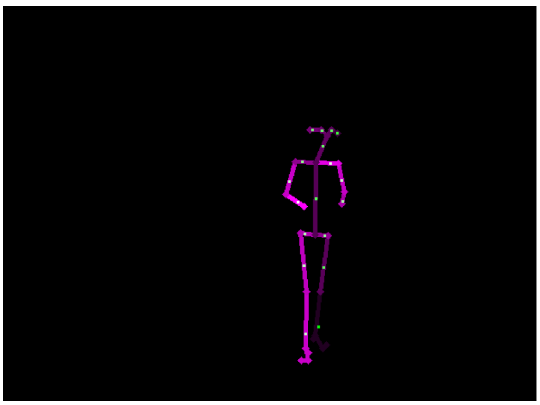

Fig. 4. Centroids of all parts are shown in green color where the skeleton is shown in pink.

After that, angle histogram is plotted from each video and every 10, 30, 60 and 90-degree data are collected to be the features. In total, there are 26 parts (18 from all body parts and 8 from enhanced arm parts) so each sequence has 26x36 = 936 features for 10 degree, 26x12 = 312 features for 30 degree, 26x6 = 156 features for 60 degree and 26x4 = 104 features for 90 degree.

2.3 KNNs for Human Action Recognition

k-Nearest Neighbors (KNNs) algorithm is a non-parametric method used for classification and regression [23]. It is mostly used for action recognition because it does not require any learning processes and it is invariant to view-point, spatial and temporal variations [18]. For classification, an object is classified by a majority vote of its *k* nearest neighbors. For example, to find out which human action a query belongs to, we use correlation distance to decide the *k* closest neighbors and $k = 3$ to 10 are used to generate the classification models. Then, a query will be assigned to a class which is the most common among those *k* neighbors.

2.4 Deep Learning

Deep learning, as a part of machine learning methods, based on artificial neural networks, has become a hotspot of big data and artificial intelligence. Deep learning is self-learning by building a multilayer model and training and evaluating the model using big amounts of data. When the data are complex and large, usually this method improves the accuracy of the classification, leading to better prediction [24]. The network architecture used is a simple network with one dense hidden layer, having 216 output units using the ReLU activation function, followed by an output layer with the softmax activation function, using a multi-class cross-entropy loss function (MCXENT) as optimization objective. To avoid the overfitting problem, we set up l2 regularization that "penalizes" the network for too large weights and prevents overfitting. The proposed solution is implemented using Waikato Environment for Knowledge Analysis (WEKA 3.8) [25]. The training parameters are introduced as follows, epochs are 10, batch size is 100 and learning rate is 0.001. 10-fold cross validation is implanted to partition the sequences into training set and testing set.

2.5 Evaluation

The UTKinect-Action 3D dataset is used in this paper and the detailed information is given in section 3.1. [1-4, 26, 27] had been tested with the same dataset and leave-one-out-cross-validation (LOOCV) was chosen to evaluate the recognition results. Therefore, to compare our method with the previous works, LOOCV is applied and the model is trained on all the sequences except the testing one.

3 Results and Discussion

3.1 Dataset

The UTKinect-Action 3D dataset [2] is composed of 10 different subjects (9 males and 1 female) performing 10 actions twice, which are *walk, sit down, stand up, pick up, carry, throw, push, pull, wave hands, and clap hands*. The dataset contains 4 parts, which are RGB images (480x640 pixels), depth images (320x240 pixels), skeleton joint locations and labels of action sequences.

A number of 199 sequences are available because one sequence (action: *carry*) is not labeled. This dataset is challenging because the length of sample actions ranges from 5 to 120 frames, occlusions of objects on the human or invisibility of body parts also have been demonstrated and the sequences are captured from different views. Since the link for downloading the description of the skeleton joint locations does not work, we cannot understand the meaning of the data in the skeleton file. Hence, OpenPose is employed to extract skeleton information. OpenPose can achieve desirable accuracy but our results with it had shown a false negative of the body in few images. This might also affect the result of our method.

3.2 Features Extraction

After getting the skeleton image from OpenPose, the optical flow of each image in the video and the centroids of each part are found and compared to collect only the orientation and magnitude information. The additional arm parts are also treated similarly for supportive differentiation in the arm area. Then, the features are collected every 10, 30, 60 and 90 degrees from the angle histogram of each video. Therefore, there are features from 18 and 26 body parts with 4 different angle histograms to compare.

3.3 Recognition Results

This section presents the principal findings of the experiment. The results obtained from analyzing the performance of KNNs and correlation distance classifiers are summarized in Table 1 and Table 2. LOOCV is applied to evaluate our method. The 198 sequences are randomly selected as training data and the one remainder is used for testing. Table 1 and Table 2 present the recognition accuracy using different k values for KNNs classifier and different collecting frequency from histogram. What stands out in Table 1 is that the highest accuracy in the dataset with 18 body parts is 77.32% (k=6 and 30 degree), whereas, 26 body parts we obtain the highest accuracy 87.98% with k=4 and 30 degree. This can be explained by the fact that 26 body parts provided more information in the arm areas that can distinguish the actions using hands like pushing and pulling better. The overall accuracy is depending on the frequency of collecting data as features. Applying 90 degree during feature extraction, less features are extracted and for this reason the accuracy is degraded in both 18 and 26 body parts. What stands out from the comparison between all feature selection methods, is the significant improvement in accuracy for the dataset with higher number of features,

except the 10 degree, where irrelevant features are presented reducing the overall accuracy of classifiers.

Table 1. The accuracy (%) of features of 18 body parts with different angle histograms.

k-nearest neighbors	Per 10	Per 30	Per 60	Per 90
k =3	65.71	75.79	71.76	63.66
k =4	68.79	76.84	74.82	62.21
k =5	69.74	74.23	73.26	65.13
k =6	70.24	**77.32**	73.29	**65.63**
k =7	70.74	74.76	73.24	62.58
k =8	71.71	75.26	**76.29**	62.08
k =9	72.71	75.24	73.26	61.58
k =10	**74.21**	77.23	75.79	61.61

Table 2. The accuracy (%) of features of 26 body parts with different angle histograms.

k-nearest neighbors	Per 10	Per 30	Per 60	Per 90
k =3	82.32	85.37	83.32	75.68
k =4	**83.37**	**87.98**	83.32	77.76
k =5	80.76	84.84	**83.84**	75.16
k =6	80.34	83.37	82.29	**78.16**
k =7	79.79	84.34	82.82	76.11
k =8	81.32	84.42	79.76	74.61
k =9	80.29	85.32	81.29	74.13
k =10	81.23	82.32	81.29	74.13

In order to analyze the accuracy for each class of the dataset, the confusion matrix of the best accuracy of each scheme is calculated and illustrated in Fig. 5. With only 18 body parts, the accuracy of *throw* class is the lowest. *Push* and *pull* classes also show misclassification between classes. This also happened with classification of *carry* class that confused with *walk* class. On the other hand, including additional arm parts in the feature using 26 body parts has improved the overall accuracy but not the accuracy of *throw* class.

Table 3. The accuracy (%) of classifying 5 human actions with features of 26 body parts every 30 degree.

k-nearest neighbors	Accuracy (%)
k =3	93.79
k =4	**96.95**
k =5	93.74
k =6	94.89
k =7	94.84
k =8	92.84
k =9	91.68
k =10	90.74

The results from our experiments are compared with previous works where the same dataset is used. Authors in [3] had chosen only 5 classes (*walk, sit down, stand up, pick*

up and carry) to train with only AAL (Active and Assisted Living) related activities and achieved 96.7% accuracy, while we reached slightly better accuracy at 96.97% (almost 97%), as Table 3 lists, with 26 body parts, 30 degree histogram collection and KNN with k=4, meaning that our method is comparable and even better. Meanwhile, the accuracy is enhanced significantly from 10 classes to 5 classes, as Table 4 lists. This improvement happens from less confusion between classes and obvious differentiation between these 5 actions.

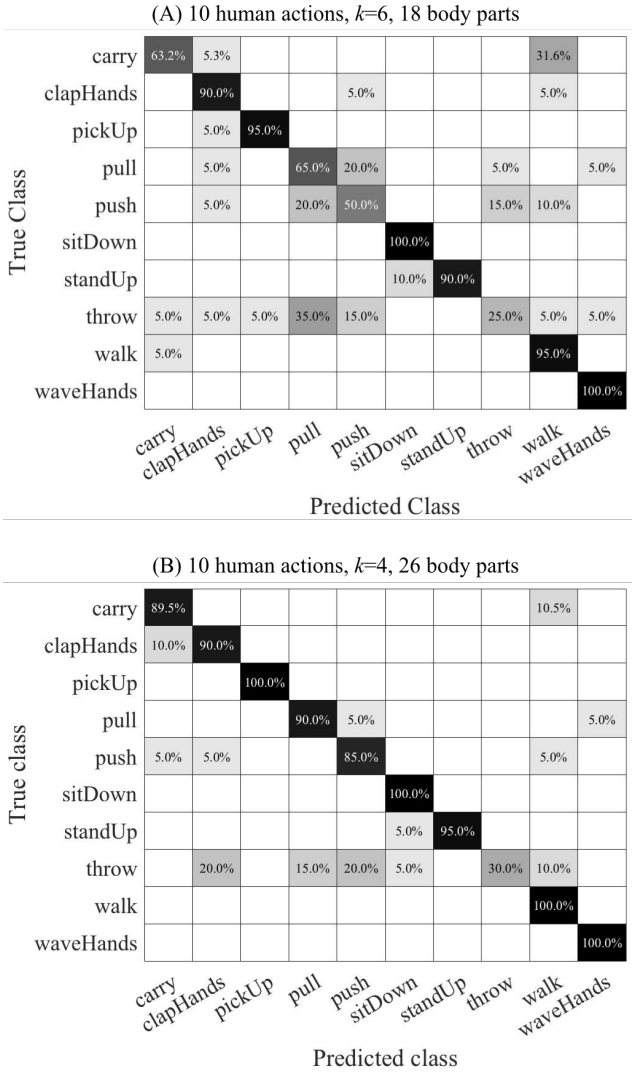

Fig. 5. The confusion matrix on the UTKinect-Action 3D dataset with the best accuracy for (A) 10 actions and 18 body parts, (B) 10 actions and 26 body parts.

These features had been used to build the model based on deep learning in order to compare the results from the proposed method as well. The results are presented in Table 4. Only 5 actions have significantly higher accuracy in both 18 and 26 body parts because the selected classes are less confused between them. With 18 body parts, deep learning performs better on 10 actions while the proposed method provides higher accuracy with 5 classes and reaches largest accuracy at 93.79%. On the other hand, 26 body parts, which include the additional information in the arm areas, performs better in all classes. The accuracy is approximately 88% even for the dataset with 10 classes, whereas KNNs performs slightly better on 5 actions dataset.

Table 4. Comparison of proposed methods based on KNNs and deep learning.

Accuracy		KNNs	Deep learning
18 body parts	5 classes	**93.79**	89.90
	10 classes	77.32	85.93
26 body parts	5 classes	**96.95**	**95.96**
	10 classes	87.98	88.44

Table 5 summarizes the results of our method and the proposed algorithms for UTKinect Action 3D dataset and LOOCV. The accuracy of all proposed algorithms are above 90%. However, our algorithm obtained 87.98% accuracy with KNNs for the dataset with 10 classes. [3] obtained the best accuracy 96.7% for 5 classes and it introduced 3D skeleton data and SVM to classify human actions. Since only 2D skeleton data is included in our algorithm, the less information is provided for training the model. Also, SVM is supervised learning for classification and regression. On the other hand, using a deep learning method for 5 classes we obtained 95.96% accuracy. The accuracy and performance of a classifier is depending on the method used for feature extraction and number of features. To accurately develop an efficient human action recognition system, different ML methods should be considered for finding the method that better fits the needs, as these systems are much related to the data set characteristics.

Table 5. Comparison of the proposed method with other ML techniques.

Algorithm	Nr. of classes	Methods	Accuracy
Our method	10	KNNs	**87.98**
Our method	5	KNNs	**96.95**
Our method	5	Deep Learning	**95.96**
Xia et al. [2]	5	Hidden Markov Models	90.9
Theodorakopoulos et al. [4]	5	Dissimilarity Space	90.95
Ding et al. [26]	10	Support Vector Machine	91.5
Jiang et al. [1]	10	Random Forests	91.9
Liu et al. [27]	10	Coupled hidden conditional random fields	92.0
Cippitelli et al. [3]	5	Support Vector Machine	96.7
Cippitelli et al. [3]	10	Support Vector Machine	95.1

4 Conclusions and Future Work

Human action recognition is a very challenging problem. The aim of this study is to introduce a solution by collecting histogram information from the optical flow of extracted skeleton data. Then, applying KNNs and deep learning methods with LOOCV on the UTKinect-Action 3D dataset. Skeleton extraction provides some false data which can lead to wrong optical flow, so we extract only from the centroid to avoid this problem and also decrease the data size. The supplementary features from arm parts have improved the accuracy remarkably. The additional data from histogram helps improve the accuracy where every 30 degree presented the best performance. Using KNNs classifier, the best accuracy obtained is 88% with k=4. However, only 5 classes display much better accuracy at about 97% with the same k value for KNN. The results are also compared with deep learning with 26 body parts. The proposed method is simpler and more efficient since those features in a sequence are characteristics to define the action, so it is not a necessity of inputting numbers of training data. The experimental results showed that by selecting the most significant features, can lead to better classification accuracy.

To improve our algorithm and classification accuracy, in the future, we will try to define the indicative position in a body part for each human action instead of taking optical flow of the centroid body part. Also, in this paper, all the features are given the same weight, we think the ideal way should be giving different weights to the features according to which body parts can exhibit the specific human action better. Finally, the SVM method will be considered for recognition.

References

1. M. Jiang, J. Kong, G. Bebis, H. Huo.: Informative joints based human action recognition using skeleton contexts. Signal Processing: Image Communication, vol. 33, pp. 29–40 (2015).
2. L. Xia, C.-C. Chen, J. K. Aggarwal.: View invariant human action recognition using histograms of 3D joints. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW ’12), pp. 20–27, Providence, RI (2012).
3. E. Cippitelli, S. Gasparrini, E. Gambi, S. Spinsante.: A Human Activity Recognition System Using Skeleton Data from RGBD Sensor. Computational intelligence and neuroscience, (2016).
4. I. Theodorakopoulos, D. Kastaniotis, G. Economou, S. Fotopoulos.: Pose-based human action recognition via sparse representation in dissimilarity space. Journal of Visual Communication and Image Representation 25(1), 12–23 (2014).
5. Z. Cao, T. Simon, S.-E. Wei, Y. Sheikh.: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. CVPR, (2017)
6. M. Wang, X. Chen, W. Liu, C. Qian, L. Lin, L. Ma.: DRPose3D: Depth Ranking in 3D Human Pose Estimation. Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, pp. 978–984 (2018).
7. A. Newell, K. Yang, J. Deng.: Stacked Hourglass Networks for Human Pose Estimation. European Conference on Computer Vision, vol. 9912, pp. 483–499 (2016).
8. A. Bulat, G. Tzimiropoulos.: Human pose estimation via Convolutional Part Heatmap Regression. European Conference on Computer Vision, vol. 9911, pp. 717-732 (2016).

9. Y. Chen, C. Shen, X. Wei, L. Liu, J. Yang.: Adversarial PoseNet: A Structure-aware Convolutional Network for Human Pose Estimation. IEEE International Conference on Computer Vision, (2017).
10. C. J. Chou, J. T. Chien, H. T. Chen.: Self Adversarial Training for Human Pose Estimation. APSIPA Annual Summit and Conference, (2018).
11. B. Mahasseni, S. Todorovic.: Regularizing long short term memory with 3d human-skeleton sequences for action recognition. In CVPR, (2016).
12. Y. Du, W. Wang, L. Wang.: Hierarchical recurrent neural network for skeleton based action recognition. In CVPR, pp. 1110–1118 (2015).
13. G. Varol, I. Laptev, C. Schmid.: Long-term temporal convolutions for action recognition. CoRR, abs/1604.04494 (2016).
14. K. Simonyan, A. Zisserman.: Two-stream convolutional networks for action recognition in videos. In NIPS, pp.568–576 (2014).
15. A. Gupta, M. S. Balan.: Action Recognition from Optical Flow Visualizations. In Proceedings of 2nd International Conference on Computer Vision & Image Processing, Advances in Intelligent Systems and Computing, vol. 703, pp 397–408 (2018).
16. L. Sevilla-Lara, Y. Liao, F. Guney, V. Jampani, A. Geiger, M. J. Black.: On the Integration of Optical Flow and Action Recognition. CoRR, abs/1712.08416 (2017).
17. S. Sun, Z. Kuang, W. Ouyang, L. Sheng, W. Zhang.: Optical Flow Guided Feature: A Fast and Robust Motion Representation for Video Action Recognition. IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2018), Salt Lake City (2018).
18. S. A. Biswas, V. N. Bhonge.: Motion based action recognition using k-nearest neighbor. International Journal of Research in Engineering and Technology, (2014).
19. J. Kaur, E. M. Kaur.: Human Action Recognition using SVM and KNN Classifiers. International Journal of Innovations & Advancement in Computer Science, vol. 5, 13–18 (2016).
20. T. Simon, H. Joo, I. Mattews, Y. Sheikh.: Hand Keypoint Detection in Single Images using Multiview Bootstrapping. CVPR, (2017).
21. S. E. Wei, V. Ramakrishna, T. Kanade, Y. Sheikh.: Convolutional pose machines. CVPR, (2016).
22. ImageJ, https://imagej.net/Introduction#What_is_Fiji.3F, last accessed 2018/11/16.
23. N. S. Altman.: An introduction to kernel and nearest-neighbor nonparametric regression. The American Statistician 43(3), 175–185 (1992).
24. K. Yu, L. Jia, Y. Q. Chen, W. Xu.: Deep learning: yesterday today and tomorrow. Journal of Computer Research and Development, vol. 50, 1799–1804(2013)
25. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I. H.: The WEKA data mining software. ACM SIGKDD Explorations Newsletter, 11(1), 10 (2009).
26. W. Ding, K. Liu, F. Cheng, J. Zhang.: STFC: Spatio-temporal feature chain for skeleton-based human action recognition. Journal of Visual Communication and Image Representation, vol. 26, 329–337 (2015).
27. A.-A. Liu, W.-Z. Nie, Y.-T. Su, L. Ma, T. Hao, Z.-X. Yang.: Coupled hidden conditional random fields for RGB-D human action recognition. Signal Processing, vol. 112, 74–82 (2015).